

Linux based 3G Multimedia Mobile-phone Reference Architecture Specification

Draft 1.0

NEC Corporation

Panasonic Mobile Communication Ltd.

Contents

Preface.....	iii
1. Introduction.....	1
2. Scope	1
3. Reference.....	1
4. Definitions and abbreviations	1
5. Architecture	3
5.1 A-CPU	3
5.2 C-CPU	5
6. Description of functional entities	5
6.1 Kernel.....	5
6.2 Common API.....	5
6.3 AP Framework	6
6.3.1 Overview	6
6.3.2 Functions	6
6.4 Telephony processing.....	8
6.4.1 Overview	8
6.4.2 Functions	8
6.5 Multi Media Processing.....	10
6.5.1 Overview	10
6.5.2 Functions	10
6.6 Data processing.....	11
6.6.1 Functions	11
6.7 Mobile Middle	11
6.7.1 Functions	11
6.8 Driver API.....	11
7. Data flow	12
7.1 Voice communication	12
7.2 Video phone.....	12
7.3 Internet Application.....	13
7.4 PPP communication for external devices	13
7.5 SMS communication.....	14

Preface

This document describes a reference architecture of Linux based 3G multimedia mobile-phone. This document is the results of the work of the CE Linux Forum's technical working group. The Architecture in this document is based on the technology which is originally the collaborative work by NEC Corporation, Panasonic Mobile Communication Ltd., and NTT DoCoMo, Inc.

1. Introduction

This purpose of this document is to define and specify overall architecture of Linux based 3G Multimedia Mobile-phone.

The basic architecture is described in chapter 5.

The functions of each component of the architecture are described in chapter 6.

The data and control flows between components are described in chapter 7.

2. Scope

This guide book defines the reference architecture of Linux based mobile phone.

It does not contain communication protocol stack.

3. Reference

[1] GTK+ API Documentation (<http://www.gtk.org/api/>)

[2] GNOME GTK+ Reference Manual (<http://developer.gnome.org/doc/API/gtk/index.html>)

[3] GNU C Library (http://www.gnu.org/software/libc/manual/html_mono/libc.html)

[4] X.org <http://www.x.org/>

4. Definitions and abbreviations

The following definitions and abbreviations are used in this document.

App	: Application programs
API	: Application Program Interface
CS	: Circuit Switched network
Engine	: Application Engine
Elib	: Extended Library
FLASH	: Macromedia Flash Player
glibc	: GNU C Library
GTK+	: The GIMP Toolkit plus
HTTP	: Hyper Text Transfer Protocol
I/F	: Interface
IR	: Infra-Red
JAM	: Java Application Manager
KVM	: K Virtual Machine
Kernel	: Linux Kernel
LCD	: Liquid Crystal Display
OBEX	: Object Exchange protocol
OCR	: Optical Character Reader
PIN	: Personal Identification Number
PS	: Packet Switched network
SD	: SD memory card
SMS	: Short Message Service

CE Linux Forum Technical Document

SSL : Secure Socket Layer
X11R6 : X Window System Release 6

5. Architecture

This document describes the architecture on the assumption that the mobile phone contains of 2 processors. One processor (A-CPU) is used for application programs. The other processor (C-CPU) is used for wireless communication.

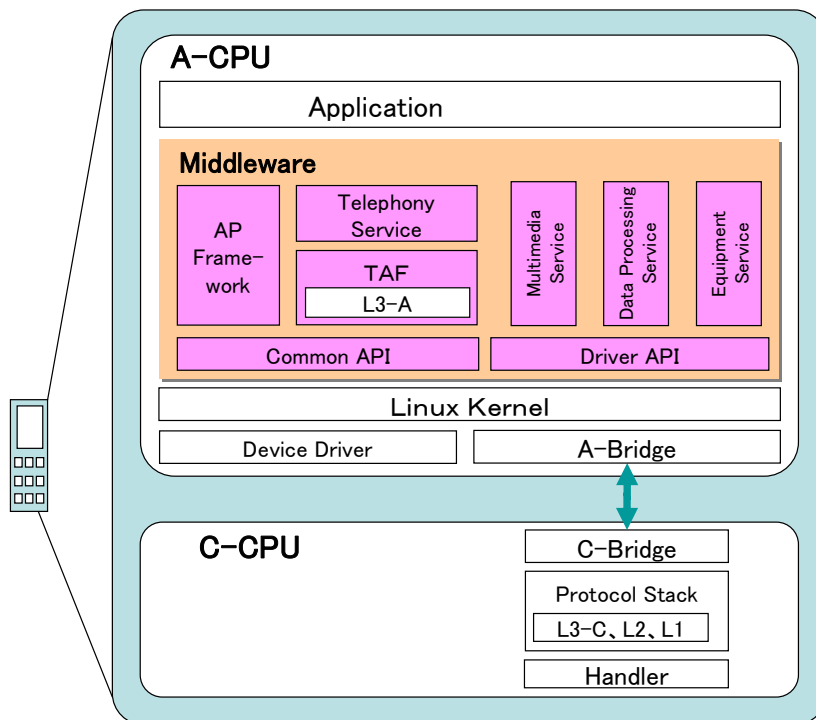


Fig-1 overall architecture

5.1 A-CPU

A-CPU contains the following 4 layers.

- Application
- Middleware
- Linux kernel
- Driver & A-Bridge

(1) Application layer

Application layer contains various applications.

They are classified into the following 8 categories.

- Telephony-applications
Telephony applications contain Standby Screen, main menu, videophone application, phone applications, phonebook, NW service and phone function setup etc.
- System applications

System-applications contain Air download, Generic LCD display, Backside LCD display, PIN authentication and monitor mode, Other function setup, Equipment alarm etc.

- Multimedia-applications

Multimedia-applications contain still image viewer, video viewer, camera app, vector graphics viewer, avatar and ring tone etc.

- Data-processing-applications

Data-processing-applications contain OCR, barcode, SD-PIM, data transfer, memory transfer, external I/F communication, user data, IR, schedule, voice memo, schedule alarm and data folder etc.

- Internet-applications

Internet-applications contain, Java AP, SMS, mailer, Browser, HTML mailer etc.

- Internet application engine

Internet application engine contains engines for HTTP, SSL, FLASH etc.

- Java application engine

Java applications engine contains JAM, class library and KVM etc.

- Others

Others contain Accessory menu, Accessories (text memo, calculator etc.).

(2)Middleware layer

Middleware layer contains the following components.

- AP framework

It provides application developers with common framework which is commonly used to develop applications for mobile phones.

- Telephony service

It provides application developers with commonly used to develop telephony applications for mobile phones.

- Multimedia service

It provides application developers with video phone service by H324.

- Data processing service

It provides application developers with processing the data from various devices, e.g. bar-code reader, optical character reader etc.

- Equipment service

It provides the functions of system management.

It conducts start and termination of system processes.

It also conducts data exchange by OBEX.

- TAF (Terminal Adaptation Function)

TAF provides its user to communication service. It consists of voice communication TAF, packet communication TAF etc.

- Common API

It provides application developers by C-language with various functions.

- Driver API

It provides middleware and application programs with device driver API.

(3)Kernel/Driver layer

Kernel / Driver layer contains Linux Kernel, device drivers and A-bridge.

Device driver contains key

A-bridge conduct communications with C-CPU.

5.2 C-CPU

C-CPU conducts wireless communication to radio access network.

Its protocol stack contains L1, L2, L3 layers. C-Bridge conducts communication with A-CPU.

6. Description of functional entities

6.1 Kernel

Linux Kernel contains following functions.

- Memory and CPU management
- Timer and system clock management
- Process management : create, destroy and dispatch
- File-system : files, directories and space management
- Console handling
- Inter-process-communication : socket
- Network communication : TCP/IP

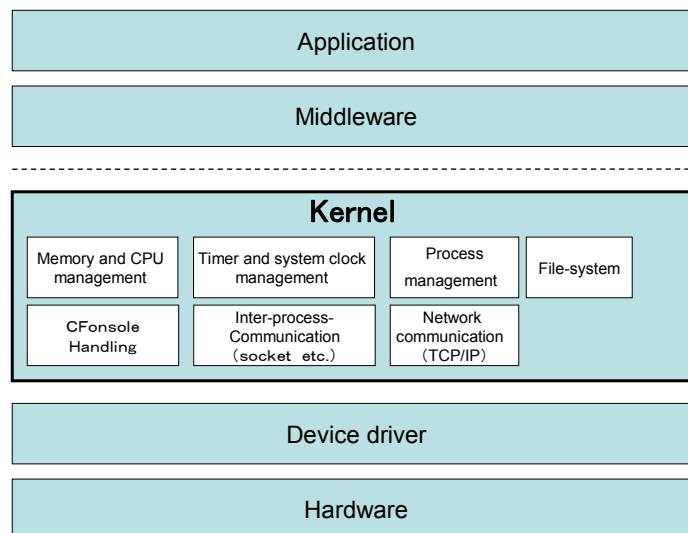


Fig-2 Linux Kernel

6.2 Common API

Common API contains various functions for applications written by C language.

Its APIs are supposed to comply with POSIX standard.

6.3 AP Framework

6.3.1 Overview

Fig-3 describes application framework overview.

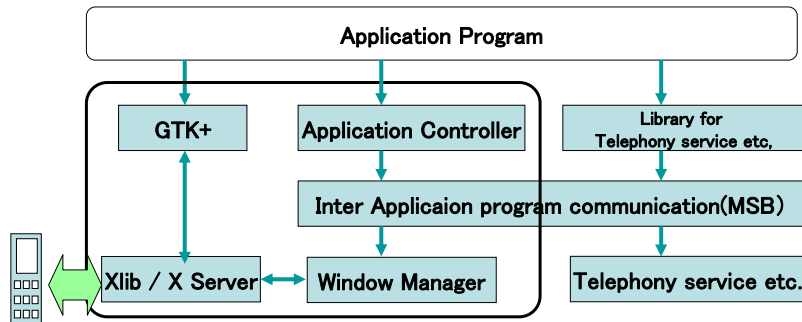


Fig-3 Application Framework

6.3.2 Functions

(1) Window Manager

In commonly used X-window system, Window Manager provides application developer with the unified operation and decoration for windows and overlap control of windows.

In the Application framework for mobile-phone, window manager conducts following functions.

- Foreground and background control of application window
- state op applications (active , in-active, idle, not-started

(2) Application Controller (APC)

APC controls the start and end of applications. That is, APC controls application start, transient-application start/end, and application end at power-off operation.

APC also manages application start status, and controls application switching operation (e.g., selection of an application to be next used as a foreground application at application switching).

WC controls window stacking, focusing, and property for each group during the period from window generation to deletion.

Window manager receives the requests from APC library through MSB and it resolves competition between applications, according to the priority table held based on the application status information.

It also control windows overlapping and key focusing by requesting X-server.

Application Controller use MSB to communicate between Application and Window Manager. At using the function of window manager application developers need not to know the functions of MSB.

(3)Xlib

Xlib is a C-language subroutine library for connecting an application program (client) to a Window System by stream connection.

(4)X-Server(X11R6)

X-Server supplies window services such as window management, event distribution, and drawing, to a client.

(5)Inter-Application program communication(MSB)

MSB supplies communication services (synchronous and asynchronous communication) between applications on this PF.

MSB provides the communication method that makes it easier for objects configured on the mobile phone to operate in coordination with one another and the system. The advantages that can be obtained by an object to use MSB are as follows:

- The communication method that is operation system- and model-independent enables an object to become both a server and a client.
- The debug feature of MSB facilitates debugging based on object linkage.
- The communication interface like RPC (Remote Procedure Call) can make synchronous communication and asynchronous communication with other objects seem to be library call.
- High intimacy with GUI Tool (GTK+, X) enables coexistence with GUI Tool.

(6)Others

PICT(PICTograph) display library

It controls turning on/off of the upper pictograph elements such as antenna and battery.

Image library

It Converts, extracts, or compresses image data, aloes used to set or acquire image information.

6.4 Telephony processing

6.4.1 Overview

Fig-5 describes application framework overview.

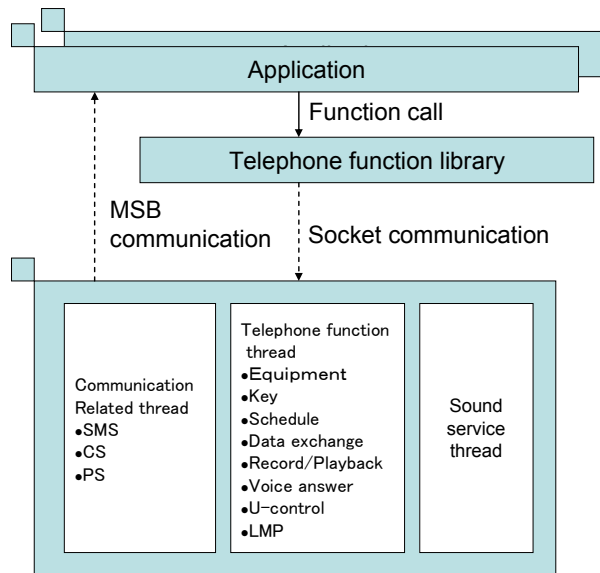


Fig-4 Telephony processing

6.4.2 Functions

(1) Packet Communication service

It provides application programs with initiation, termination, rejecting incoming call etc. for packet communication service.

(2) Voice Communication service

It provides application programs with dialing, call disconnection, rejecting incoming call etc. for voice communication service.

(3) SMS Communication service

It provides application programs with notification of events and status of SMS service etc.

(4) Equipment service

It provides application programs to setup various mode of equipment and reference status of the equipment.

(5) Authentication/DRM

It provides application programs with authentication/DRM service.

(6) Schedule

It provides application programs to register a schedule, sort the schedule data, read to-do data etc.

(7) Data Exchange

It provides application programs with the functions to handle phone-book memo image and video etc.

(8) Record and Playback

It provides application programs with record and playback of voice memo.

(9) LMP Management

It provides application programs with functions for lamp of the equipment.

(10) Sound System

It provides application programs with functions for ring-tone and melody of the equipment.

(11) Resource Management Library

It provides application programs with setting and reference of owner's and registered phone number, name and e-mail addresses.

(12) SD File

It provides application programs with the functions for SD files, e.g. read, save, delete etc.

6.5 Multi Media Processing

6.5.1 Overview

Fig-6 describes application framework overview.

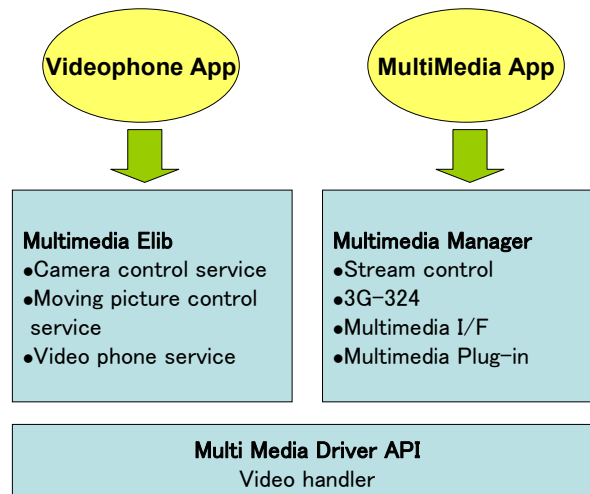


Fig-5 Multimedia

6.5.2 Functions

(1)Multimedia Manager

Multimedia Manager provides interfaces for SCA (moving picture control and camera control), multimedia functions, video phone ELIB, 3G-H324M, camera control ELIB, and moving picture control ELIB.

(2)Multimedia Elib

Multimedia Elib provides interfaces for camera control, moving picture control, and video phone services.

(3)Multimedia Driver API

Multimedia Driver API provides video handler interfaces.

6.6 Data processing

6.6.1 Functions

(1)Bar Code library

It provides the bar code reader functions.

(2)OCR library

It provides the OCR access reader functions.

6.7 Mobile Middle

6.7.1 Functions

(1)OBEX module

Defined are the specifications (OBEX interface thread) for providing an interface that is used by OBEX to perform communication processing based on request messages from the high-order APL and return processing results to the high-order APL and the specifications (OBEX Versit library) for providing an interface that is used by OBEX to convert objects via the Versit library.

(2)MAW service

Described are the specifications for the interface between programs relating to operation status monitoring tasks (MAW/ADM). As operation status monitoring tasks, MAW and ADM mainly monitor the activation of each task at the time of power on and the deactivation of each task at the time of power off, as well as the charging and other statuses of the mobile terminal.

6.8 Driver API

Driver API provides upper layer components (middleware and application programs). It is useful for the developer to developing middleware and application programs for mobile phones.

7. Data flow

In this chapter, data and control flow between components of the architecture are described.

7.1 Voice communication

Telephony application controls C-plane by telephony service.

Voice data is outputted to speaker through AMR codec.

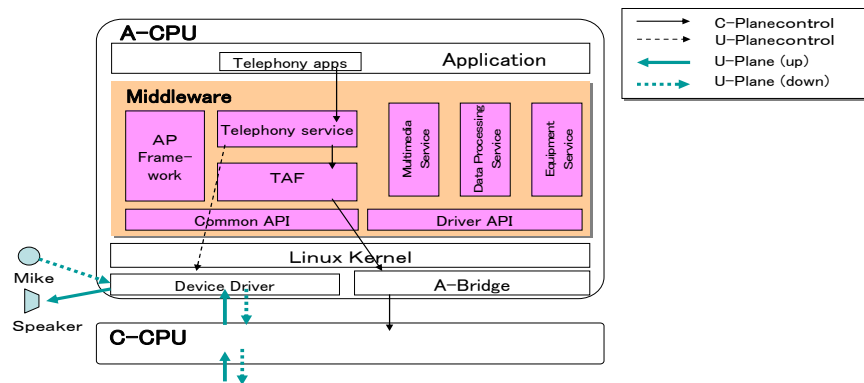


Fig-6 Voice Communication

7.2 Video phone

Video phone application program controls C-Plane by telephony service.

Multimedia service conducts video-phone service by H.324.

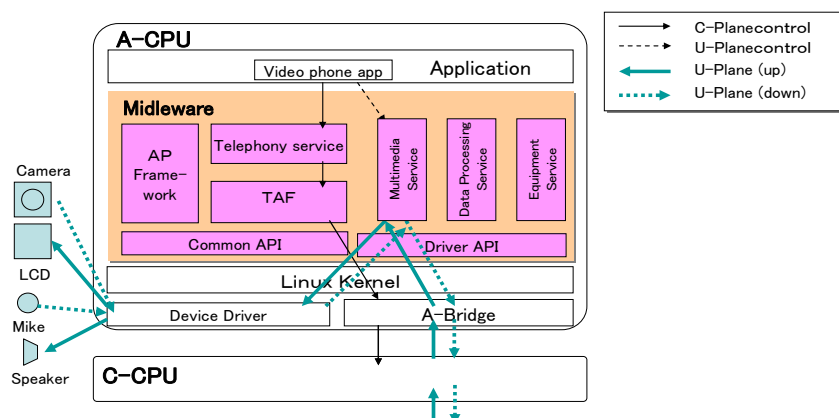


Fig-7 Video Phone

7.3 Internet Application

Internet Application program controls C-Plane by telephony service.

Data from the Internet is transported by TCP/IP in Linux Kernel.

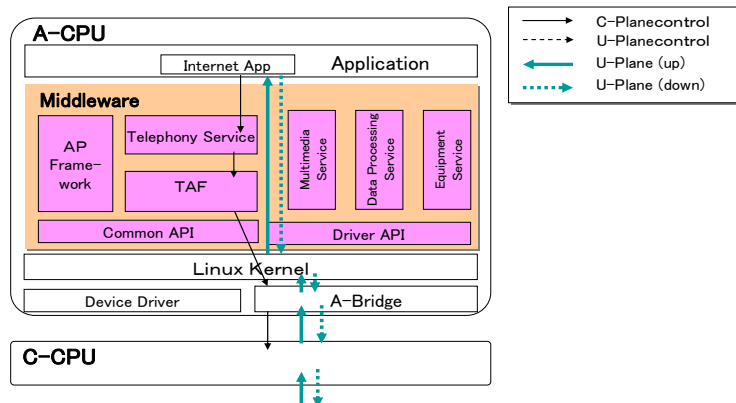


Fig-8 Internet Application

7.4 PPP communication for external devices

Applications program for external communication controls C-Plane by telephony service.

PPP communication data are transported to external devices through USB driver.

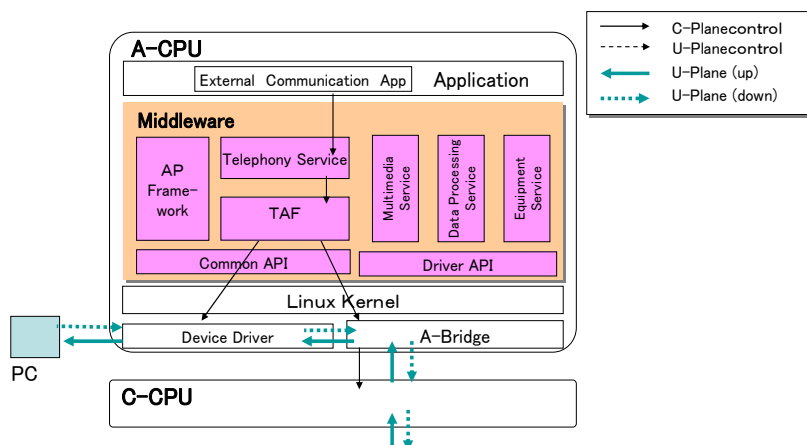


Fig-9 PPP Communication for external devices

7.5 SMS communication

Telephony Service controls C-Plane and U-Plane.

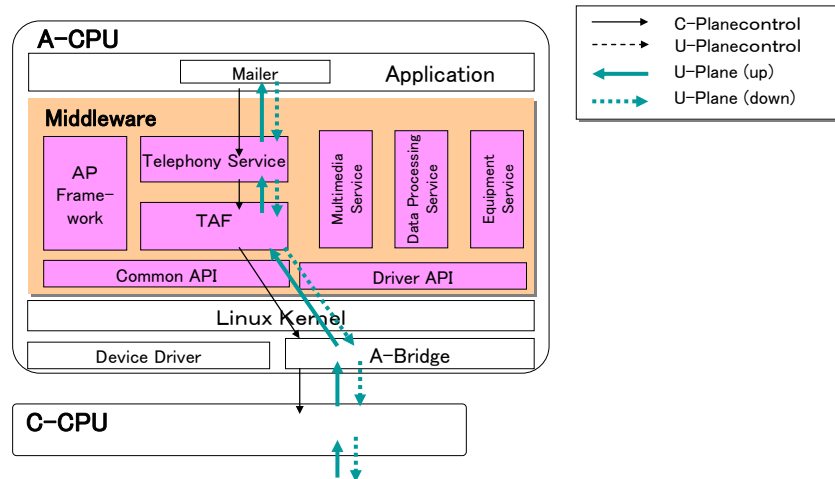


Fig-10 SMS Communication